Project Milestone-- Data Storage Implementation: KV + relational

Faazil Shaikh - 100707829

1. **Watch the first three videos for Kafka connectors (focus on the concepts, not the details) from https://www.confluent.io/blog/kafka-connect-tutorial/**

2. Describe the following:

- **Sink and Source connectors.**

A sink connector sends data through Kafka topics to external systems, which are typically databases.

A source connector is a component that collects data from a system. Databases can be used as source systems.

- **The applications/advantages of using Kafka Connectors with data storage.**

Advantages of Kafka connectors is that data from external systems can be imported into Kafka topics, and data from Kafka topics can be exported to external systems

- **How do Kafka connectors maintain availability?**

It can be used in a distributed mode allowing for fault tolerance and horizontal scaling, which, of course, improves availability.

- **List the popular Kafka converters for values and the properties/advantages of each.**

Avro Converter, Protobuf converter and Json Schema Converter is popular

- Avro Converter works with schema registry to move data in and out of Avro format for Kafka Connect.
- Json schema defines the overall structure of JSON data and the converter allows it to convert data for consuming kafka topics
- Protobuf is similar to XML but faster, the converter works with the registry to convert the data in and out of the format.

3. **Search the internet to answer the following question:**

- **What's a Key-Value (KV) database?**
  - A key-value database is a type of non relational database that uses basic key-values to store data.
  - In a key-value database, data is maintained as a set of key-value pairs, with a key serving as a unique identifier.
  - Keys and values can be any sort of object.
- **What are KV databases' advantages and disadvantages?**

Advantages:

- Easy to use
- Scalable
- Speed
- Reliability

Disadvantages:

- Needs a single key and value

- **List some popular KV databases.**

- Oracle NoSQL
- Amazon DynamoDB
- ScyllaDB
- Azure Cosmos DB
- Heroku Redis

4. **Follow the following videos to deploy and use Redis and MySQL databases using GKE.**

https://drive.google.com/file/d/1c_QsBePZqTU-PeiuLLLdEHiCNE9bimsn/view?usp=sharing

5. **Follow the following video to set up sink and source Kafka connectors to the deployed MySQL database.**

https://drive.google.com/file/d/1YsKxMUrZ7RjMYKQqAcA5PhVydmWFKdzS/view?usp=sharing

6. **Follow the following video to set up a Kafka connector to the deployed Redis database.**

https://drive.google.com/file/d/1c_QsBePZqTU-PeiuLLLdEHiCNE9bimsn/view?usp=sharing

7. **Now, you will store a dataset into cloud storage. The dataset has to be sent into Kafka topics and connectors have to be configured to automatically store the dataset into the data storage. The producer that will send the dataset to Kafka topics should run on your local machine as it will simulate real sensors while Kafka, connectors, and data storage should be on the cloud. Use MySQL for the CSV files and Redis for images. Feel free to update the Yaml files from the given repository to fit your dataset.**
8. **Use the sensors, images, ground truth Pose in the lastest session in http://robots.engin.umich.edu/nclt/ as your dataset. Record a video showing the configuration of Kafka connectors, producers' python script, a proof of successfully stored data into data storage.**

   See videos above

9. **List some possible applications that can be implemented by using the uploaded dataset.**
   - Light detection and ranging
   - Geo-mapping
   - Object Detection